**Pràctica 3**

**Variables**

Una variable és un dipòsit on s’emmagatzema un valor. Consta d'un nom i pertany a un tipus (numèric, cadena de caràcters, etc.).

Una variable pot emmagatzemar:

* Valors Numèrics (100, 260, 1.24, 2.90, 5.00, etc.)
* Cadenes de caràcters ( "Joan", "Compres", "Llistat", etc.).
* Valors lògics (true, false).
* Hi ha altres tipus de variables que veurem més endavant.

Les variables són noms que posem als llocs on emmagatzemem la informació. En JavaScript, han de començar per una lletra o un subratllat (\_).

Una variable no pot tenir el mateix nom d'una paraula clau del llenguatge (Paraules reservades). En JavaScript tenim les següents paraules reservades:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Paraules reservades de Javascript** | | | | |
| abstract | delete | function | Null | throw |
| boolean | do | goto | Package | throws |
| break | double | if | private | transient |
| byte | else | implements | protected | true |
| case | enum | import | public | try |
| catch | export | in | return | typeof |
| char | extends | instanceof | short | var |
| class | false | int | static | void |
| const | final | interface | super | while |
| continue | finally | long | switch | with |
| debugger | float | native | syncronized | true |
| default | for | new | this |  |

**Declaració de variables**

Una variable es defineix anteposant la paraula clau var:

var dia;

Es poden declarar diverses variables en una mateixa línia:

var dia, mes, any;

A una variable se la pot definir i immediatament inicialitzar amb un valor:

var edat = 20;

o, si no en dos passos:

var edat;  
edat = 20;

**Tipus de variables**

Encara que totes les variables de JavaScript es creen de la mateixa manera (mitjançant la paraula reservada var), la forma en què se'ls assigna un valor depèn del tipus de valor que es vol emmagatzemar (nombres, textos, etc.)

1. **Variables numèriques**  
     
   S'utilitzen per emmagatzemar valors numèrics enters (anomenats integer en anglès) o decimals (anomenats float en anglès). En aquest cas, el valor s'assigna indicant directament el nombre enter o decimal. Els nombres decimals utilitzen el caràcter. (Punt) en comptes de, (coma) per separar la part entera i la part decimal:  
     
   var iva = 16; // variable tipus sencer  
   var total = 234.65; // variable tipus decimal
2. **Cadenes de text**  
     
   S'utilitzen per emmagatzemar caràcters, paraules i / o frases de text. Per assignar el valor a la variable, es tanca el valor entre cometes dobles o simples, per delimitar el seu començament i el seu final:  
     
   var missatge = "Benvingut al nostre lloc web";  
   var nomProducte = 'Producte ABC';  
   var lletraSeleccionada = 'c';
3. **Arrays**

De vegades, als arrays se'ls anomena vectors, matrius i fins i tot arranjaments. No obstant això, el terme array és el més utilitzat i és una paraula comunament acceptada en l'entorn de la programació.  
  
Un array és una col·lecció de variables, que poden ser totes del mateix tipus o cadascuna d'un tipus diferent. La seva utilitat es comprèn millor amb un exemple senzill:

si una aplicació necessita gestionar els dies de la setmana, es podrien crear set variables de tipus text:  
  
var dia1 = "Dilluns";  
var dia2 = "Dimarts";  
...  
var dia7 = "Diumenge";  
  
Tot i que el codi anterior no és incorrecte, sí que és poc eficient i complica en excés la programació. Si en comptes dels dies de la setmana s'hagués de guardar el nom dels mesos de l'any, el nom de tots els països del món o els mesuraments diàries de temperatura dels últims 100 anys, s'haurien de crear desenes o centenars de variables .  
  
En aquest tipus de casos, es poden agrupar totes les variables relacionades en una col·lecció de variables o array. L'exemple anterior es pot refer de la següent manera:  
  
var dies = [ "Dilluns", "Dimarts", "Dimecres", "Dijous", "Divendres", "Dissabte", "Diumenge"];  
  
Ara, una única variable anomenada dies emmagatzema tots els valors relacionats entre si, en aquest cas els dies de la setmana. Per definir una matriu, s'utilitzen els claudators per delimitar el seu començament i el seu final i s'utilitza el caràcter, (coma) per separar els seus elements:  
  
var nombre\_array = [valor1, valor2, ..., valorN];  
  
Un cop definit una matriu, és molt senzill accedir a cadascun dels seus elements. Cada element s'accedeix indicant la seva posició dins de l’array. L'única complicació, que és responsable de molts errors quan es comença a programar, és que les posicions dels elements comencen a comptar en el 0 i no en l'1:

var diaSeleccionat = dies[0]; // diaSeleccionat = "Dilluns"

var altreDia = dies[5]; // altreDia = "Dissabte"

En l'exemple anterior, la primera instrucció vol obtenir el primer element de l'array. Per a això, s'indica el nom de l'array i entre claudàtors la posició de l'element dins de l’array. Com s'ha comentat, les posicions es comencen a comptar en el 0, de manera que el primer element ocupa la posició 0 i s'accedeix a l'mitjançant dies[0].

El valor dies[5] fa referència a l'element que ocupa la sisena posició dins l’array. Com les posicions comencen a comptar a 0, la posició 5 fa referència al sisè element, en aquest cas, el valor ‘dissabte’.

1. **Booleans**

Les variables de tipus boolean o booleà també es coneixen amb el nom de variables de tipus lògic i el seu funcionament bàsic és molt senzill.

Una variable de tipus boolean emmagatzema un tipus especial de valor que només pot prendre dos valors: true (vertader) o false (fals). No es pot utilitzar per emmagatzemar nombres i tampoc permet guardar cadenes de text.

Els únics valors que poden emmagatzemar aquestes variables són true i false. A continuació es mostra un parell de variables de tipus booleà:

var clientRegistrat = false;

var IVAinclos = true;

Durant el curs veurem la seva utilitat.

**Elecció del nom d'una variable**

Hem de triar noms de variables representatius. En l'exemple anterior els noms dia, mes, any són prou clars per donar-nos una idea sobre el seu contingut, una mala elecció de noms hagués estat cridar-les a, b i c. Podem donar-li altres bons noms. Altres no són tan representatius, per exemple d, m, a. Possiblement quan estiguem codificant un programa ens semblin noms representatius, però passat un temps l’oblidaríem.

**Impressió de variables en una pàgina HTML**

Per mostrar el contingut d'una variable en una pàgina utilitzem l'objecte document i cridem a la funció write. En el següent exemple definim una sèrie de variables i les anem imprimint:

<!DOCTYPE html>

<html>

<head>

<title>Pràctica 3</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<script>

function inici(){

var nom = 'Joan';

var edat = 10;

var alçada = 1.92;

var casat = false;

document.write('<h1>Impressió de variables en una pàgina HTML</h1>');

document.write('<br>');

document.write(nom);

document.write('<br>');

document.write(edat);

document.write('<br>');

document.write(alçada);

document.write('<br>');

document.write(casat);

}

window.onload=inici;

</script>

</head>

<body>

</body>

</html>

Quan imprimim una variable, no l'hem de disposar entre cometes (en cas de fer això, apareixerà el nom de la variable i no el seu contingut).

Els valors de les variables en les que emmagatzemem noms (és a dir, són cadenes de caràcters) han d'anar tancades entre cometes. Els valors de les variables numèriques (en aquest exemple les variables edat i alçada) no han d'anar tancades entre cometes. Cada instrucció finalitza amb un punt i coma.

Les variables de tipus boolean poden emmagatzemar només dos valors: true o false.

# 

# El resultat en visualitzar la pàgina ha de similar a aquest:

# Impressió de variables en una pàgina HTML

Joan  
10  
1.92  
false

És a dir que es mostren els continguts de les 4 variables. Una variable és d'un tipus determinat quan li assignem un valor:

Per mostrar el contingut d'una variable en una pàgina hem d'utilitzar la funció 'write' que pertany a l'objecte document. Recordem que el llenguatge Javascript és sensible a majúscules i minúscules i no serà el mateix si escribim: Document.write (nom);

**Exercici**

1. Confeccionar una programa en JavaScript on haureu de definir una variable del tipus cadena de caràcters per desar el nom d’un treballador i una altra variable del tipus numèrica per desar el seu sou. Imprimir cada variable en una línia diferent a la pantalla.
2. Defineix una array amb els mesos de l’any i imprimeix només els mesos imparells.